SOFTWARE ARCHITECTURE DESIGN FOR MICROSATELLITE COMPUTER BOARD USING SYSML LANGUAGE

N. Levineț. M.Sc., V. Ilco, M.Sc., A. Gîrșcan, PhD student,
N. Secrieru, PhD, Associate Prof.
Technical University of Moldova

INTRODUCTION

The term "*On Board Computer*" indicates, rather obviously, any unit flying on board a satellite which provides processing capability. For a satellite, board computer controls all subsystems, as positioning system, communication subsystem, power system and payload (camera). To manage with a satellite systems usually is used a microcontroller which is programmed. The program is necessary for supervision and monitoring resolved tasks by of all peripheral systems. To develop a program with high reliability is required to design an overall architecture. This structure is divided per blocks to highlight tasks and to detect errors that may appear in software development process.

Detection and correction of errors at the planning stage will reduce software development time essentially. In some cases to correct errors in a program, is necessary to change the subprograms and operation logic. The number of modification cycles depends on the programmer experience and on the algorithms planning correctness. Therefore, it is rational to design algorithms in the form of diagrams, and based on these diagrams to design board computer program or any embedded system.

The diagrams can be produced through an artificial language Unified Modeling Language (UML) for describing models and software specifications. UML was developed to represent object-oriented programming, whose purpose consists in structuring classes and instances (called objects). For software development of computer board are created diagrams reflecting the requirements of the each system. The diagrams represent the operating principles of the software for board computer of the satellite.

After modelling class and sequence diagrams, design environment allows automatically to create activity diagrams. These diagrams will be changed according to the running algorithms. The activity and class diagrams will enable the generation of "*body*" of the program. The program will be completed with the appropriate syntax and finally reverse design can be achieved, based on software designed automatically update diagrams.

Documentation and diagrams submitted electronically will allow to detect the error in a shorter time. After correcting the error, using design environment will be possible to regenerate the program code. The technical documentation designed diagrams can be used in the testing stage, and validation phase.

1. TASKS OF THE BOARD COMPUTER

Board computer is autopilot that is leading satellite systems. The main functions of the computer is to transmit commands to peripheral devices and process control to achieve these commands.

To charge batteries it is necessary to position the satellite toward the sun. And for capturing and sharing images, positioning takes place towards the earth station antennas. Depending on the location of the satellite, the computer transmits positioning coordinates to Attitude Determination and Control System (ADCS). Positioning data are stored in a memory board computer. Capture system is connected only when the satellite is in the mapping zone. Board computer connects capture system and send command that the image have to be transmitted to earth station.

To receive data from any earth station, the telemetry must be transmitted without interruption. Board computer, in case that the batteries reach the minimum level of stored energy, stops this process. Therefore, at different stages of operation of the satellite in orbit, board computer transmits commands and receives responses after considering whether the task was executed correctly.

On board computer memory is stored grid of commands and responses for each system. To send commands is necessary to establish an algorithm selection and transmission of commands under the communications protocol. Next algorithm analyzes fairness of orders execution. If an error occurs, the computer attempts to resolve this situation.

These errors with the telemetry collected from satellite devices are stored in memory by the board computer. In the below image are presented tasks that are realized by board computer. At architectural design we have a task description as SYSML diagrams.

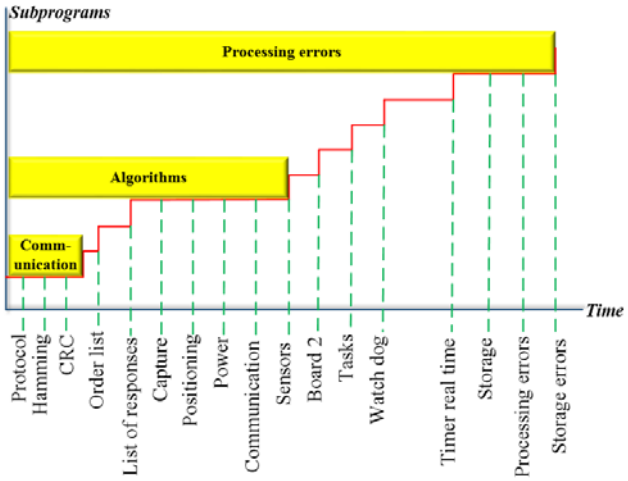

Figure 1. Diagram of algorithms for board computer.

According to algorithms generated by UML diagrams, board computer operates these functions at different stages depending on orbital position of satellite.

2. BOARD COMPUTER SOFTWARE ARCHITECTURE

For each algorithm is developed SysML diagrams that reflect the operation of each task. In Figure 2 is shown diagram that presents the process of selection and transmission commands from stored grid. The list of functions is developed based on technical documentation of each satellite system. The search of a function is made following an algorithm and after this data is extracted in a temporary buffer. These data will be used to carry out the message that will be sent to the system. Since processing method of responses is similar with functions generation architecture, it is reasonable to modify existing diagrams. This data is processed

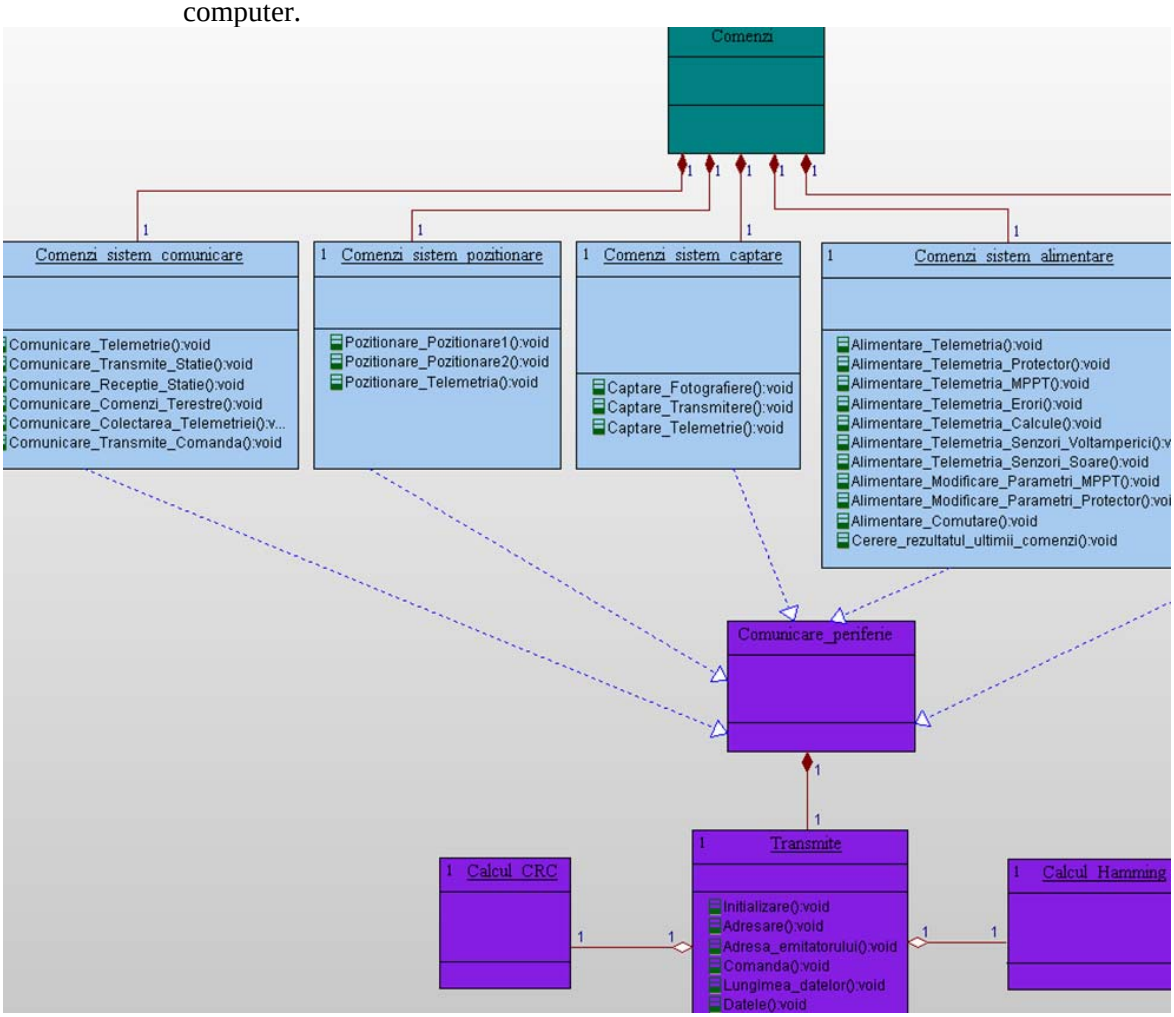

Figure 2. Class diagram for satellite system commands.

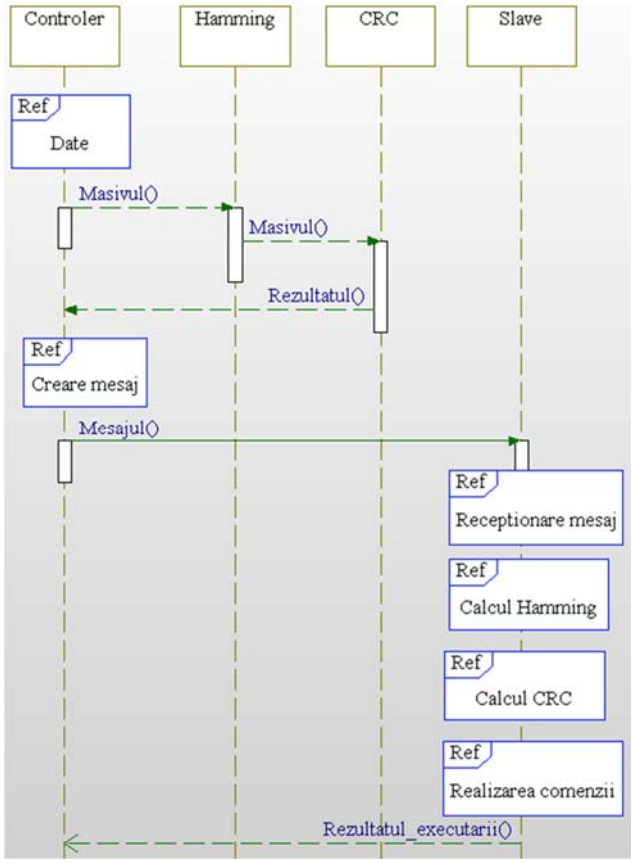

Figure 3. Sequence diagram for onboard computer communication satellite systems.

according to designed algorithms based on operating methods for each system. Therefore, other diagrams may represent some diagrams, in the case that encapsulation occurred. Telemetry is collected from the peripheral systems.

Based on structure diagrams shall be designed activity diagrams. After this will be developed program for board computer.

At the stage of drafting structure diagrams, is no way to see the execution state of a task at a time. Therefore, the next step is to develop sequence diagrams and state diagrams.

From class diagrams, we found that communication and system work algorithms would be most called subprograms.

Communication between the onboard computer and satellite systems is achieved according to a established protocol. Send or received data are sent to the Hamming error-correcting algorithm and to the checksum calculate algorithm. These data are encapsulated in the final message. The algorithms return variables, that will describe the fairness of received message. If errors are detected, the computer will send a non-acknowledgment message. After this, ”slave” resends the message. In figure 4 is shown the process for the control of received data.

According to the activity diagram, is developed software that will be implemented on MSP430 family controller. When data is received via UART interface, byte shall be entered in a unidimensional array, and data counter is incremented. If the counter exceeds the length of transmitted data, the data is

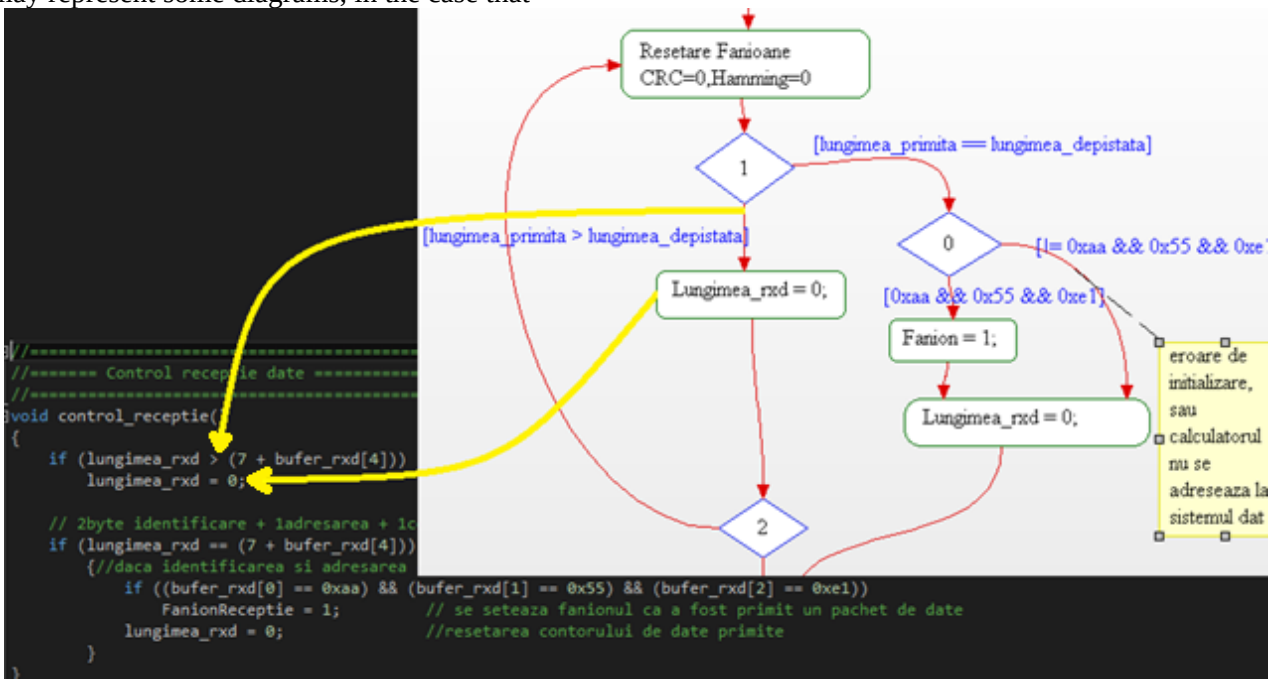

Figure 4. Control packet reception subprogram.

neglected and counter resets. The reception flag is set after checking several conditions.

The length of the data must be equal to the counter. Initialization and addressing must match the system address. If the flag is set in ”1”, the data packet moves forward for data reception checking and eventually data correction.

Board computer software stores telemetry and errors collected from peripheral systems.

Statistics of errors occurred is required to determine the operation of the satellite in space, and to correct operating parameters of the satellite. The correction of parameters is achieved by changing variables from programs for each block.

Errors may occur in communication processes, at the watchdog reset, or malfunctioning of each module. Telemetry is collected from peripheral modules, after this it is found the average acquired magnitudes at different moment of time.

3. CONCLUSIONS

SysML diagrams are needed to model the structure and behavior of the software concept. Due to algorithms developed at the initial design stage of the program, we obtain a reliable structure and minimum code length. After detecting errors, using reverse design engineering, the diagrams can be regenerated based on the correct program.

Therefore, electronic documentation is changed automatically immediately. Documentation is necessary to exchange ideas with members of a team, or teams that designs peripheries "slaves". If it is required, the documentation can be used to design other types of systems, considering that first diagrams will provide a general overview of functional blocks.

Therefore, for software architecture design of the power supply system of the satellite were used structure diagrams of onboard computer. That minimized design time a lot.

Making the block diagrams lead to restricting of problems and focus attention on some issues.

Actuality of the problem of developing software architecture is expressed by the advantages offered by this method:

1. Reduce design time and system validation testing.

2. Increasing software reliability.

3. Generating documentation automatically.

4. Exchange of ideas between team members. The thoughts are clearly expressed using the diagram that are evident in terms of intuitive. Understanding program time for another person is shorter, because we spend time just to understand diagrams instead of program code;

5. It emphasizes the weakness of the system and it delimits system problems. As the program is divided into blocks, there is a concentration of attention on certain aspects of the details without taking into consideration other problems. Therefore,

from the ”chaos” of questions and problems, step by step is approached software architecture.

6. Due to instrumentation, the designer returns to previous diagram and has the opportunity to make the necessary changes.

7. Body program generating is possible by structure diagrams. If the program code is structured, then there is a decrease in the number of errors.

8. Thanks to reverse engineering, it is possible to modify and improve documentation automatically. That will decrease the test time.

9. After validation based on electronic documentation can be performed technical support. It means that using SysML diagrams it is possible to change functioning of the system.

10. Using the current architecture for developing other types of systems. This advantage will decrease design time for other systems.

References

1. Wesley A. *Enterprise Patterns and MDA - Building Better Software with Archetype Patterns and UML”. 528 page. Pub Date: December 22, 2003.*

2. Wieringa R.J. *Design Methods for Reactive Systems: Yourdon, Statemate, and the UML. 457 page. Morgan Kaufmann Publishers 2003.*

3. Delligatti L*. SysML Distilled: A Brief Guide to the Systems Modeling Language. 265 page. Paperback – November 18, 2013.*

4. Kima Cheol-Han, Westonb R.H., Hodgsonb A., Lee Kyung-Huy. *The complementary use of IDEF and UML modelling approaches.*

5. Eickhoff J. *Onboard Computers, Onboard Software and Satellite Operations, 282 p.*

6. Solvhoj J. *Onboard Computer for Pico Satellite. Technical University of Denmark January, 63 pp. 2002.*

7. Kuwahara T. *FPGA-based Reconfigurable On-board Computing Systems for Space Applications. 200 pp. October 29, 2009.*

8. Larman C. *Applying UML and patterns. An Introduction to Object-Oriented Analysis and Design and the Unified Process. Second edition. 616 pp. 2002.*

Recommended for publication: 29.04.2015.